

We claim:

1 1. A method for scheduling a resource to service a plurality of pending requests received
2 from a plurality of schedulable entities, while preventing each schedulable entity from exceeding
3 a maximum quality of service allocated to each schedulable entity, comprising:

4 selecting a request associated with a schedulable entity;

5 responsive to determining that servicing the selected request will exceed the
6 schedulable entity's maximum quality of service, advancing a virtual time for
 scheduling the requests, without servicing the request; and

 responsive to determining that servicing the selected request does not exceed the
 schedulable entity's maximum quality of service, servicing the request and
 advancing the virtual time.

2. The method of claim 1, wherein the request includes a request to allocate disk space.

3. The method of claim 1, wherein the request includes a request to allocate memory.

1 4. The method of claim 1, wherein the request includes a request for network bandwidth.

1 5. The method of claim 1, wherein the request includes a request for CPU processing
2 cycles.

1 6. The method of claim 1, wherein the request is selected using a fair-share scheduling
2 algorithm.

1 7. The method of claim 6, wherein the fair-share scheduling algorithm is a weighted fair-
2 share scheduling algorithm, each weight corresponding to a schedulable entity's minimum
3 quality of service allocation.

1 8. The method of claim 7, wherein the minimum quality of service allocated to each
2 schedulable entity is a minimum percentage share of the resource.

1 9. The method of claim 6, wherein the fair-share scheduling algorithm is a hierarchical
fair-share scheduling algorithm.

10. The method of claim 6, wherein the fair-share scheduling algorithm is a hierarchical
weighted fair-share scheduling algorithm, each weight corresponding to a schedulable entity's
minimum quality of service allocation.

11. The method of claim 6, wherein the fair-share scheduling algorithm is a start-time
fair queuing algorithm with virtual time scheduling.

1 12. The method of claim 1, wherein each request includes a requested duration, the
2 method further including:

3 limiting the requested duration of the request to a pre-determined request duration
4 upper bound.

1 13. The method of claim 1, wherein the maximum quality of service allocated to each
2 schedulable entity is a maximum percentage share of the resource.

1 14. The method of claim 1, wherein a rate controller determines if servicing the request
2 will exceed the schedulable entity's maximum quality of service.

1 15. The method of claim 14, wherein if the rate controller determines that servicing the
2 request will exceed the schedulable entity's maximum quality of service, the request remains
3 pending.

1 16. A method for scheduling resource requests from a plurality of schedulable entities,
2 wherein each resource request includes a requested duration and each schedulable entity has a
3 maximum quality of service guarantee, the method comprising:

4 assigning a start number tag to a resource request using a start-time fair queuing
5 algorithm with virtual time scheduling;

6 selecting the resource request with the smallest start number tag, the selected request
7 having an associated schedulable entity;

8 limiting the requested duration of the selected resource request to a pre-determined
9 duration upper bound;

10 servicing the selected resource request if servicing the selected resource request will
11 not exceed the associated schedulable entity's maximum quality of service
12 guarantee; and

13 advancing a virtual time value.

1 17. The method of claim 16, further including:

2 updating the start number tag for a resource request associated with the schedulable
3 entity that made the selected resource request if the selected resource request
4 is not serviced.

1 18. The method of claim 16, further including:

2 leaving the selected resource request pending if servicing the selected resource
3 request will exceed the schedulable entity's maximum quality of service
4 guarantee.

19. A system for scheduling pending resource requests from a plurality of schedulable
2 entities while limiting a maximum quality of service allocated to each schedulable entity,
3 comprising:
4

5 a plurality of schedulable entity queues for holding pending resource requests, each
6 schedulable entity queue holding resource requests from a schedulable entity;

7 a scheduler for selecting resource requests from the plurality of schedulable entity queues
8 using a fair-share scheduling algorithm, and further adapted to increment a virtual time value
9 each time a resource request is selected; and

10 a plurality of rate controllers associated with the plurality of schedulable entity queues,
11 each rate controller adapted to limit the rate at which resource requests selected by the scheduler
are serviced to the schedulable entity's maximum quality of service.

1 20. The system of claim 19, wherein each rate controller is further adapted to:

2 monitor the servicing of resource requests from the rate controller's associated
3 schedulable entity queue to calculate the quality of service received by the
4 schedulable entity; and

5 block the servicing of a selected resource request if the schedulable entity's maximum
6 quality of service would be exceeded if the selected resource request was
7 serviced.

21. The system of claim 19, wherein each schedulable entity queue is associated with a
weight, and the scheduler uses a weighted fair-share queuing algorithm.

22. A hierarchical system for scheduling resource requests from a plurality of child
schedulable entities while limiting the maximum quality of service allocated to a plurality of
parent schedulable entities, comprising:

a plurality of child schedulable entity queues for holding pending resource requests, each
5 child schedulable entity queue holding resource requests from a child schedulable entity;

6 one or more child schedulers for selecting resource requests from the plurality of child
7 schedulable entity queues using a fair-share scheduling algorithm, and further adapted to
8 transmit selected resource requests to a parent schedulable entity queue;

9 a plurality of parent schedulable entity queues, each parent schedulable entity queue
10 receiving resource requests from a subset of the child schedulable entity queues, each parent
11 schedulable entity queue holding resource requests received from one of the child schedulers;

12 a parent scheduler for selecting resource requests from the plurality of parent schedulable
13 entity queues using a fair-share scheduling algorithm, and further adapted to increment a virtual
14 time value each time a resource request is selected; and

15 a plurality of rate controllers associated with the plurality of parent schedulable entity
16 queues, each rate controller adapted to limit the rate at which resource requests selected by the
17 parent scheduler are serviced to a parent schedulable entity's maximum quality of service.

1 23. A computer program product for scheduling a plurality of pending requests for
2 service from a resource received from a plurality of schedulable entities, while preventing each
3 schedulable entity from exceeding a maximum quality of service allocated to each schedulable
4 entity, the computer program product comprising:

a computer readable medium that stores program code including:

program code that selects a request associated with a schedulable entity using a
fair-share scheduling algorithm;

program code that services the request if a rate controller determines that
servicing the request will not exceed the associated schedulable entity's
maximum quality of service; and

program code that advances a virtual time in the fair-share scheduling algorithm.

1 24. The computer program product of claim 23, wherein the fair-share scheduling
2 algorithm is a weighted fair-share scheduling algorithm, each weight corresponding to a
3 schedulable entity's minimum quality of service allocation.

